A PROJECT REPORT ON

ONLINE BOOK STORE

Book Store Web
Application to Purchase
Books

SUBMITTED IN PARTIAL

FULFILMENT OF

PG DIPLOMA IN ADVANCED COMPUTING

(PG-DAC)

UNDER THE GUIDANCE OF
Mr. Swaraj Choudhary

PRESENTED BY

230340120055	Shruti Chintawar
230340120198	Shreyas Dhule
230340120032	Anup Kanere
230340120006	Adarsh Kadam
230340120158	Pranal Ghule

AT

CENTRE FOR DEVELOPMENT OF ADVANCED

COMPUTING C- DAC, PUNE

ACKNOWLEDGEMENT

The project "Online Book Store" was a great learning experience for us and we are submitting this work to Advanced Computing Training School (C- DAC ACTS, Pune).

We are very glad to mention the name of Mr. Swaraj Choudhary for his valuable guidance to work on this project.

We are highly grateful to Ms. Risha P. R., Manager of ACTS Training Centre, CDAC, for her guidance and support whenever necessary during the course of our journey to acquire PG-Diploma in Advanced Computing (PG-DAC) through CDAC ACTS, Pune.

Our heartfelt thanks go to Mrs. Namrata Ailawar (Course Coordinator, PG-DAC) who gave us all the required support and kind coordination to provide all the necessities to complete the project and throughout the course up to the last day of the course.

From:

230340120055	Shruti Chintawar
230340120198	Shreyas Dhule
230340120032	Anup Kanere
230340120006	Adarsh Kadam
230340120158	Pranal Ghule

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

The project's main goal is to develop an online book store where users can search for and make purchases of books based on title, author, and subject. The user's chosen books are displayed in a tabular format, and they can order them online using a debit card or credit card. Instead of going to a bookstore and wasting time, the user can purchase a book on this website.

Many online book retailers, such as MPHOnline and BookDepository were developed using HTML. I'd like to create a comparable website with.NET and SQL Server.

An online book store is a web application that allows customers to buy books online. Customers can search for a book by title or author using a web browser, add it to their shopping cart, and then purchase it using a debit or credit card transaction. The user can log in using his or   her account information, or new customers can swiftly create an account. They should include their full name, phone number, and shipping address. A user can also provide feedback to a book by rating it on a scale of one to five. The books are organised into several categories based on the subject matter, such as Action and Adventure, Classics, Fantasy, Fiction, Non-fiction, Horror etc.

Customers can shop online for books at the Online Book Store using a web browser. A customer can create an account, sign in, add items to a shopping cart, and make purchases using his or her debit/credit card information.

When compared to a regular user, the Administrator will have more features. The Administrator can add, delete, and edit book data, and member information, as well as confirm an order that has been placed. The Online Book Store is built using the Master page, data sets, data grids, and user controls.

Chapter 1: INTRODUCTION

1.1 Background of Study

COVID-19 is one of most serious crises in modern history, and it came as a shock to medical professionals, governments, businesses, and individuals all across the world. Many countries and politicians were adopting bold and strict measures to stop the virus from spreading further and causing the healthcare system to collapse. Many businesses and industries are experiencing major problems as a result of people being forced to stay at home and practice social distance. The spread of digital technologies and the internet, on the other hand, allows humans to connect and communicate and businesses to interact with their consumers despite being physically separated. With the advancement of technology, brands and merchants may continue to sell their products to customers and conduct at least some of their company operations. Businesses who approach this challenging time of building a strong digital presence with a long-term strategy will have a much better chance of thriving in the post-COVID-19 era.

The reason why I selected online book store web services is because everyone must exercise social distancing, which requires spending the most of the day at home. The objective of this project is to develop an e-book store where people may buy books from the comfort of their own homes. A virtual book store on the internet where customers can browse the catalogue and select books of interest is known as an online book store. A shopping cart can be used to collect the selected books. The items in the shopping cart will be displayed as an order at the moment of checkout. More information will be required at that time to complete the transaction. The customer will generally be required to fill out or select a billing address, a shipping address, a shipment option, and payment information such as a debit or credit card number. As soon as the order is placed, the customer receives an email notification.

1.2 Problem Statement

Companies all over the world are being impacted by the Covid-19 pandemic. Some of the world's largest corporations experienced unfavorable consequences, such as empty stores due to a lack of customers. The COVID-19 epidemic, on the other hand, has encouraged the growth of the online shopping convenience movement. Starting with an easy-to-use website, businesses may make it simple for customers to conduct research, compare alternatives and specifications, and do anything else relevant online.

This project aims to develop an online shopping for customers with the goal so that it is very easy to shop loved things from an extensive number of online shopping sites available on the web. The customer simply requires a PC or a laptop and one important payment sending option to shop online. To get to this online shopping system, all the customers will need to have an email and password to login and proceed to their shopping. Upon successful login, the customer can purchase a wide range of books. The customer will pick their favorite books from the online book store sites. They do not need to go physical shops, instead just need a computer and a payment making options like net banking, credit card or debit card. If an order has not yet been shipped out to the customer, the customer may cancel it. The system cancels the order with the publishers or returns the books to them in the occurrence of a cancellation. A book can also be returned within a week. The payments will be credited to the customer's debit or credit card, depending on their preference.

1.3

1.3.1 Main Objective

An online bookstore software project that serves as a central database for all of the books in stock, as well as their title, author, and price. The goal of this project is to create a website that serves as a central book store. This website was built with php on the front end and SQL on the back end. Various book-related details are stored in the SQL database. A user visiting the website will find a wide selection of books organized by category. The user can choose a book and see its price. The user can even utilize the website to look for certain books. After the user chooses a book, he/she must fill out a form before the book is booked for the user.

Customers may shop for books online using a web browser thanks to the Online Book Store Project. A customer can create an account, log in, sort books by category, add books to a shopping basket, and pay their bill using their credit card information. When compared to a regular user, the Administrator will have more options. He can edit the author, publisher, book categories, book details, and member information, as well as confirm an order.

The following are the three main components of the software:
1. Implementation of a new user registration and login process.
2. Allow the user to select any book.
3. Allow the user to purchase books.

Specific Objectives

- To design an online book store management system.
- To analyze the problems in the existing system.
- To analyze the possible requirements for the new system.

1.1 Purpose and Scope of Study

1.1.1 Purpose

The purpose of an online shopping system would be to achieve the following goals:

- Create a web user interface for adding, viewing, and deleting records in different areas.
- Create a user interface for inputting computer details.
- Provide a user interface for changing computer and accessory details.
- Provide a user interface that allows users to browse the store and select things to purchase.

1.1.2 Scope

The main scope deliverables of the project would be to:

- Analyze and develop detailed specifications and requirements
- Prepare high-level and detailed system design specifications
- Prepare a test plan as well as test cases.
- Develop the system and write the code.
- Unit, integration, and system testing should all be performed.
- Demonstrate a bug-free application after making any necessary changes.

Chapter 2: LITERATURE REVIEW AND THEORY

2.1 Literature Review

When we use Google to search for educational websites and applications, we will find a lot of options. However, there is some ambiguity in selecting suitable content at the proper time. Some websites have been developed that contain stories, novels, essays, and other types of content. Electronic Commerce (e-commerce) applications allow multiple parties involved in a commerce transaction to connect in order to shop for new, secondhand, rare, and out-of-print books. Review of literature for an online bookstore. The prototype serves as a roadmap for establishing a solid Online Book Ordering System based on user feedback, notably from the perspective of academics, which will be handled by the university's book store. The arranging, however, is still done manually.

According to a software development firm, an e-catalogue delivers vital information about product specifications to potential customers. It makes it easier for potential customers to find the items they want in the format they want. It is the ideal and ideal product catalogue because it does not require printing, has no number or color restrictions, and does not require distribution. The system is described as self-updatable in a few simple clicks, with the content in the e-catalogue always being accurate and, best of all, requiring no reprinting. Furthermore, the e-catalogue promotes the products on its own, with greater interactivity, consumer personalization, and even a shopping cart for inquiries.

MPHOnline, BookDepository, and Amazon are few of the most popular online shopping sites that employ an e-catalogue to display their products. The website lists the many types of books that are available. The book cover can be found on the left .

description, which includes information such as the title, author, price, and number of volumes left to be sold, as well as customer ratings.

2.1 Existing System

Figure 1: MPHOnline

Strength

Within 3 to 5 business days, the customers can anticipate their orders to arrive. The website also sells stationery, office products, toys, and games in addition to books. You can search for a book by title, author, ISBN, or publisher using the prominent search bar at the top of the website.

Weakness

Shipment to West Malaysia is free for non-members and RM80 for members on orders above RM120. The company charges RM7 per shipment for orders that are less than the specified amount. Free shipping is provided for orders over RM120 for non-members and RM80 for members in East Malaysia. Orders over a certain amount apply for free shipping and no additional handling fees. The company charges RM12 each shipment for orders under the stipulated amount, plus an extra RM3 handling cost per book (both members and non-members).

Figure 2: BookDepository

Many Malaysians are unaware of Book Depository, an online book retailer that is a hidden gem. This online book store is situated in the United Kingdom, but it ships worldwide, including to Malaysia. Even better, the site provides free international shipping.

Strength

The Book Depository is a true find. This online book store is situated in the United Kingdom, but it ships worldwide for free. They have over 19 million titles available for sale at cheap costs. Keywords, best sellers, 'quick movers,' 'book of the week,' and other search options are available on the website. It's a great place to browse for those hard-to-find books while still saving money on shipping.

Weakness

The main disadvantage of purchasing books from this website is that delivery can be slow. Regardless of whether the book is shipped from their UK or Australia center, it can take anywhere from 7 to 14 days (often up to 3 weeks) for an order to arrive in Malaysia.

Chapter 3: METHODOLOGY

3.1 Research Methodology

The first step in starting this project is gathering all of the necessary requirements in order to develop a website that is easy to maintain and meets user needs. The emphasis is primarily on study into various online bookstores that are already operational. Accordingly, questionnaires and interviews with target end users have been shown to be the most effective methods for gathering exact data from them. The design paradigm starts with a prototype of the graphical user interface (GUI) layer and then outlines the website's back end (database and system architecture) system, which is only visible to the portal's administrators. The prototype's construction was adjusted in response to end-user feedback in order to build an online book catalogue that meets the project's requirements at the completion. Iteration occurs as the prototype is refined to meet the user's needs while also allowing the developer to gain a better understanding of what needs to be done. The website is then deployed on a web hosting to make it available via the internet after the GUI design prototype is finalized.

Within a software project, the Software Development Life Cycle (SDLC) is a process that is followed for a software project. It is a detailed strategy that describes how to build, maintain, replace, and change or improve particular software. The life cycle is a mechanism for enhancing software quality and the development process itself. During the software development process, numerous software development life cycle models have been established and designed. To assure success in the software development process, each process model follows a set of stages specific to its type. Agile Methodology is the model I'm using for this project.
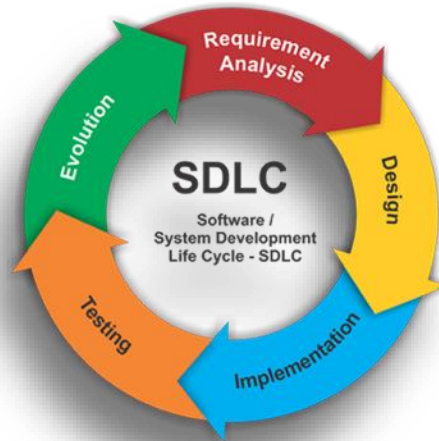
Figure 4: Illustration of SDLC

Agile Methodology

To deliver a smooth execution of the project, a methodology has been chosen to assist in managing the project phase by phase, beginning with the initiation and ending with the closure. A methodology is necessary for selecting the best methods, practices, approaches, and procedures from among the many available. As a result, Agile Methodology was chosen for the development of the Online Book Store System platform. Agile technique is the ideal strategy to complete this project because it divides it into phases and requires constant adjustments and input until the final product is completed.

Agile methodology is an iterative project management strategy that allows a project to adapt to changing working conditions and develop in a short amount of time. In this method, which is referred to as "sprint," a project term is divided into shorter and repeatable stages. Sprint lengths were decided during the project's early planning stages and will be reflected in the final product.
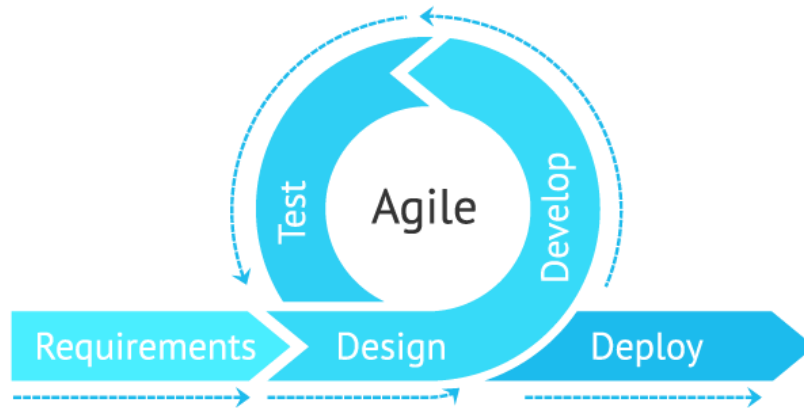
Figure 5: Illustration of Agile Methodology

3.1 Process

3.1.1 Research, Analysis and Gathering Requirement

This phase includes conducting literature reviews and analyses of existing systems in order to determine the requirements and features for an online book system website. Prepare documentation after evaluating the research study. Surveys and feedback are used to gather requirements. Requirements are subject to change over time. The prerequisites for the Online Book System platform are listed below:

1) A product catalogue on the home page

   After a successful login, the user will be directed to this page. It will provide all of the book categories and provide a search keyword option for finding the desired title. It also has some unique elements, such as recommended titles and weekly special books.

2) Search Search textbox provides the user with the option of searching by keyword. The book title should be used as the keyword.

3) Advanced Search

   The user can utilize advanced search to find a book based on the title, author, category, and price range. The total number of books that match the specific search

criteria will be presented. The user can choose a book and add it to their shopping cart from here.

4) Book Description

 If a user wants more information about a book, he/she can click on the title and be taken to a Book Description page. It includes annotations on the book's content as well as a link to Amazon.com where you can read the book review.

5) Voting by users

A user can rate a book based on his or her interests. They can give it a five-star rating if it's excellent, four if it's very good, three if it's good, two if it's regular, and one if it's deficient. The final rating of a book is determined by the sum of all user ratings.

6) Shopping Cart

The user can manage a shopping cart that contains all of the books that they had chosen. The user can change, delete, and update the contents of their shopping cart. A final shopping cart summary is shown, which includes all of the products selected by the user as well as the overall cost.

7) Managing User Account

To access all of the website's functions, each user should create an account. The user can log in and out using the login and logout pages, respectively. The database will save all of the user sessions.

8) Administration

Special functions for the Administrator will be granted, such as adding or deleting a book category, adding or deleting a member, managing member orders, adding or deleting a debit/credit card type, and so on.

3.1.1 Planning Phase

Project Tasks/Weeks	1	2	3	4	5	6	7	8	9	10	11	12
Idea & Title Proposal	■	■										
Introduction		■										
Aim & Objective			■	■	■							
Analysis & Survey					■	■	■					
Methodology							■	■				
Proposal Defense							■	■	■	■		
Design & Development										■	■	■
Interim Report Submission												■

Table 1: Gantt Chart for FYP 1

Project Tasks/Weeks	1	2	3	4	5	6	7	8	9	10	11	12
Design Phase	■	■	■	■								
Prototype Development		■	■	■	■							
Prototype Testing & Maintenance			■	■	■	■	■	■				
Prototype Finalization & Documentation								■	■	■	■	■
Dissertation Submission												■
Viva Presentation												■

Table 2: Gantt Chart for FYP 2

3.1.1 Project Design Phase

In order to design a web site, the relational database must be designed first. The data model and the process model are the two aspects of conceptual design. The data model specifies which data should be stored in the database, whereas the process model defines how the data should be handled. To put it another way, the data model is used to construct the relational tables of a relational database. The process model is used to design the queries that will access those tables and perform operations on them.
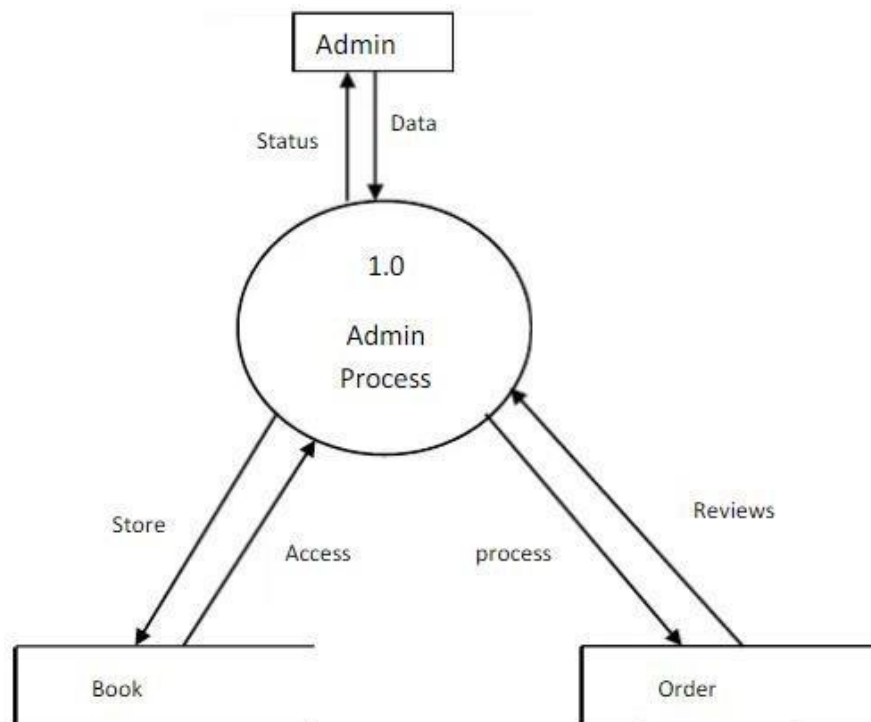
A decomposition diagram illustrates a system's top-down functional decomposition and highlights its structure. The goal of the Functional Decomposition is to break down a system step by step, starting with the system's main function and progressing through the intermediate levels to the level of basic functions. More extensive process diagrams, such as Data Flow Diagrams (DFD) start with this diagram. Data Flow Diagrams illustrate the flow of data from external entities into the system, as well as from one process to the next. A DFD can be drawn using four symbols:

    i.   Rectangles that represent external entities such as data sources and destinations.

    ii.   Ellipses, which represent processes that receive data as input, validate, process, and output it.

    iii.   Data flows are represented by arrows, which can be electronic data or physical items.

    iv.   Data stores, especially electronic stores such as databases, are represented by open- ended rectangles or a Disk symbol.
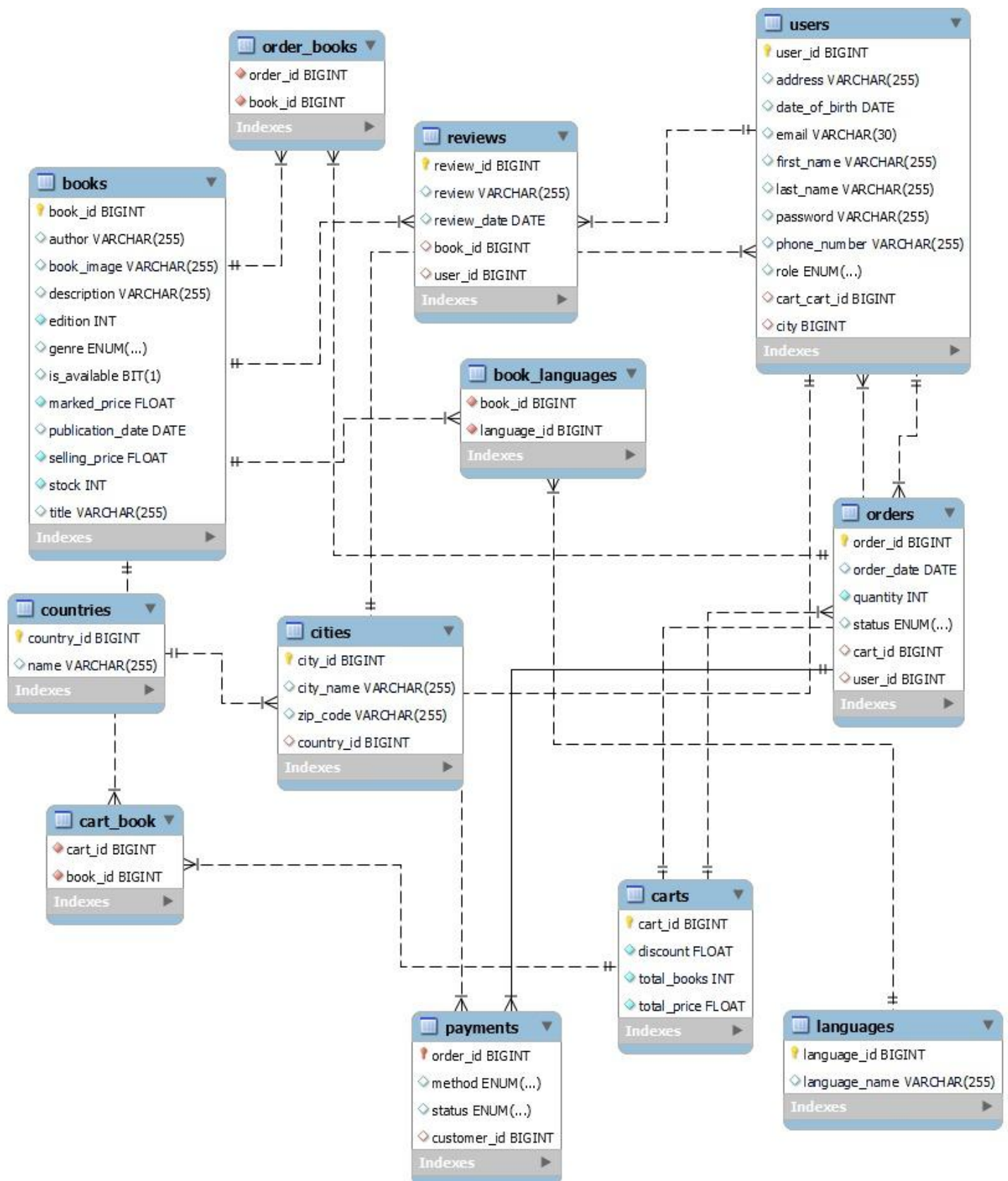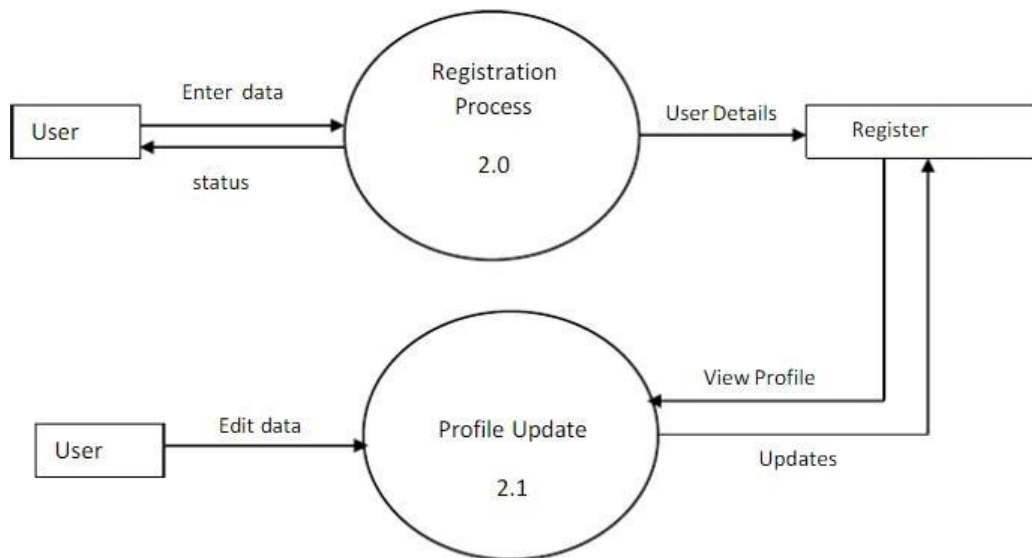
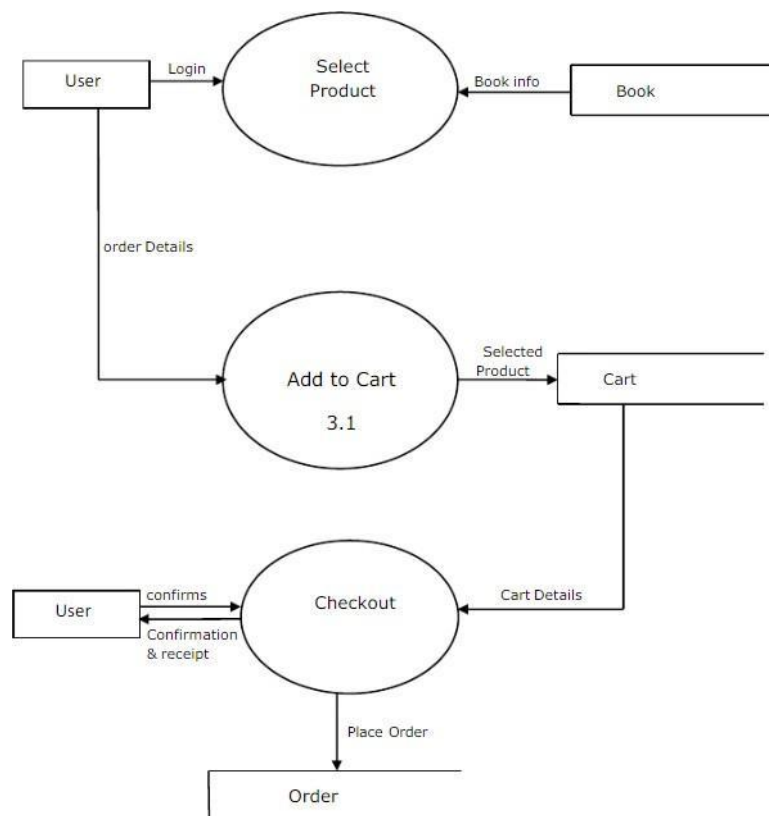0-Level DFD:

DFD for Admin Process:

ER Diagram :

order_books
- ◆ order_id BIGINT
- ◆ book_id BIGINT
- Indexes ▸

books
- 🔑 book_id BIGINT
- ◇ author VARCHAR(255)
- ◇ book_image VARCHAR(255)
- ◇ description VARCHAR(255)
- ◇ edition INT
- ◇ genre ENUM(...)
- ◇ is_available BIT(1)
- ◇ marked_price FLOAT
- ◇ publication_date DATE
- ◇ selling_price FLOAT
- ◇ stock INT
- ◇ title VARCHAR(255)
- Indexes ▸

reviews
- 🔑 review_id BIGINT
- ◇ review VARCHAR(255)
- ◇ review_date DATE
- ◇ book_id BIGINT
- ◇ user_id BIGINT
- Indexes ▸

users
- 🔑 user_id BIGINT
- ◇ address VARCHAR(255)
- ◇ date_of_birth DATE
- ◇ email VARCHAR(30)
- ◇ first_name VARCHAR(255)
- ◇ last_name VARCHAR(255)
- ◇ password VARCHAR(255)
- ◇ phone_number VARCHAR(255)
- ◇ role ENUM(...)
- ◇ cart_cart_id BIGINT
- ◇ city BIGINT
- Indexes ▸

book_languages
- ◆ book_id BIGINT
- ◆ language_id BIGINT
- Indexes ▸

orders
- 🔑 order_id BIGINT
- ◇ order_date DATE
- ◇ quantity INT
- ◇ status ENUM(...)
- ◇ cart_id BIGINT
- ◇ user_id BIGINT
- Indexes ▸

countries
- 🔑 country_id BIGINT
- ◇ name VARCHAR(255)
- Indexes ▸

cities
- 🔑 city_id BIGINT
- ◇ city_name VARCHAR(255)
- ◇ zip_code VARCHAR(255)
- ◇ country_id BIGINT
- Indexes ▸

cart_book
- ◆ cart_id BIGINT
- ◆ book_id BIGINT
- Indexes ▸

carts
- 🔑 cart_id BIGINT
- ◇ discount FLOAT
- ◇ total_books INT
- ◇ total_price FLOAT
- Indexes ▸

payments
- 🔑 order_id BIGINT
- ◇ method ENUM(...)
- ◇ status ENUM(...)
- ◇ customer_id BIGINT
- Indexes ▸

languages
- 🔑 language_id BIGINT
- ◇ language_name VARCHAR(255)
- Indexes ▸

DFD for User Registration and Profile Update:
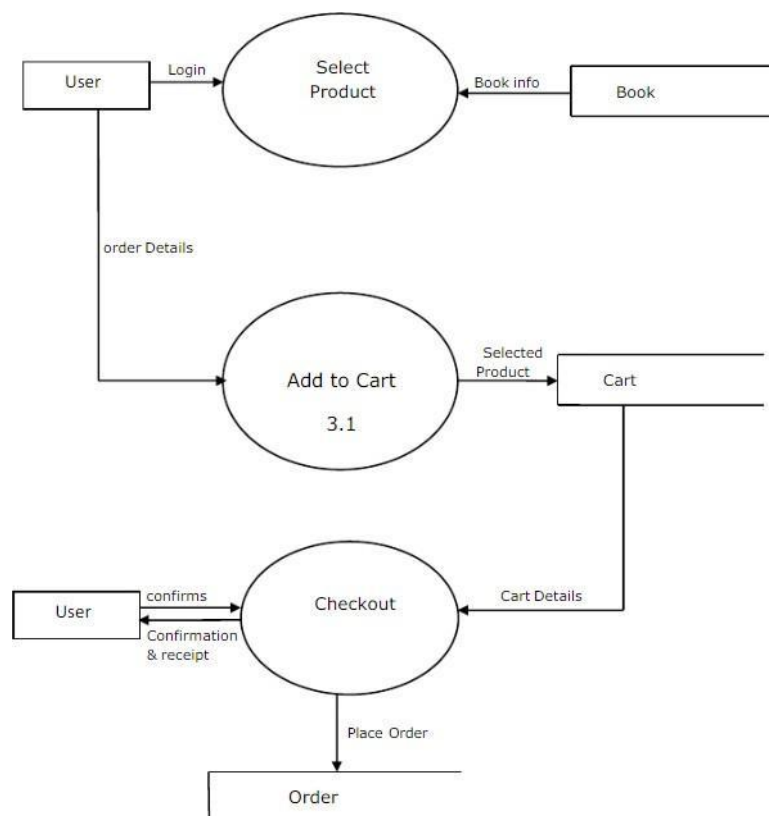

DFD for Shopping and Checkout Process:

3.1.1 Development, Testing and Review Phase

Development Phase includes writing codes using programming languages. The website will be programmed using HTML and CSS. In the meantime, MySQL will be used as the database. This process will be repeated until the client is satisfied with the final product.

Maintaining Phase test the website's functionality. This process will take place in parallel with the development phase. This is to check that the programmed codes are working properly and that there are no errors in the coding.

Review Phase includes obtaining feedback from users and testers. This process will be followed once the finalized website has been created. This process is necessary in order to develop the platform in the future. For the evaluation process, this website will be shared with a group of users or testers.

3.1.1 Development, Testing and Review Phase

Development Phase includes writing codes using programming languages. The website will be programmed using HTML and CSS. In the meantime, MySQL will be used as the database. This process will be repeated until the client is satisfied with the final product.

Maintaining Phase test the website's functionality. This process will take place in parallel with the development phase. This is to check that the programmed codes are working properly and that there are no errors in the coding.

Review Phase includes obtaining feedback from users and testers. This process will be followed once the finalized website has been created. This process is necessary in order to develop the platform in the future. For the evaluation process, this website will be shared with a group of users or testers.

Chapter 4: DESIGN

4.1 Introduction

The design phase of the system development process determines the best solution for the problem that was found during the analysis phase. The suggested system's comprehensive blueprint is produced at the end of the design phase. This chapter discusses popular software development processes and the proposed system, alternative solutions, design techniques, and the Web-Based Book Store System overall design.

4.2 Development Strategy

The development strategy is the path to follow in order to meet the requirements that have been identified. In terms of the Web Book Store System, numerous strategies have been considered. Purchasing a set of commercial software can decrease development costs and time while also providing more quality, but you may have to pay for certain needless features, it may not match all of your requirements precisely, and it is not configurable. Standalone systems have advantages such as being simple to set up and maintain, having fewer incompatibilities, and having less security risks, but they are unable to handle remote users.

The Online Book Store System chose to construct a web-based system from scratch as its development strategy. When software is built from scratch, the end result is software that is exactly aligned to the defined requirements. In addition, a highly configurable and upgradeable solution with few incompatibility difficulties is created. The ability to serve remote users via a web-based system is critical in order to meet needs such as allowing clients to place orders, searching for books, and tracking order status.

The well-defined methods for the purpose of system design and modelling are known as design techniques. Each one has its own set of techniques, features, advantages, and disadvantages. As a result, it should be carefully studied before deciding on a method.

4.3 Why chose the Web-Based System

- From any location, it's simple to keep track of stock processes and provide useful data.
- Because the database is centralized and everything is synchronized, maintenance is simple.
- It is simple to set up.
- Client-server architecture with a web browser as the client interface is possible.
- The user is merely redirected to the proper pages.
- Because model-view-controller has been adapted, future updates and maintenance are simple.
- By implementing a web-based solution, developer can save money on both hardware and software.
- Web systems are platform independent, meaning it can run on any platform.

4.4 Design of the Online Book Store System

In order to make the design process and subsequent execution more efficient, Online Book Store Web-Based System is organized into five parts.

- Purchase Module
- Stock Module
- Financial Module
- Web Module

- Administration Module

Each module's top-level use case diagram is illustrated in the sub sections below. Case narratives are used in some situations, and additional diagrams are provided as needed.

4.4.1 Purchase Module

Only the administrator has access to this module. Requisitions, purchase orders, purchase, purchase return orders, and purchase returns are all covered in this section. Admin manages requisitions and purchase orders as completed or cancelled. Before creating a new purchase order, the admin examines the pending requisitions, checks their availability, and if everything is in order, they can add it as a new book and place an order. Because the details provided by the customer may vary and the availability of the agent is uncertain, we cannot use the requisitions automatically for the purchase order procedure. They should be confirmed before being added as a new book.

Case name	Manage purchase orders
Actors	Admin
Description	Admin manages the purchase orders
Pre-conditions	Admin must log in to the system
Main flow	1. Actor navigates to the 'Purchase' section. 2. System displays the section. 3. System provide options to add, complete and cancel purchase orders. 4. System updates and confirms.
Post-conditions	None

4.4.2 Stock Module

This module contains three sections: Admin - the organization that supplies the books, Books - the book details, and Stock - the stock details. Stock keep unit (SKU) is a standard term for product ID, a unique code used to identify products in inventory control. The current status determines the availability of functionalities such as edit and terminate.

Case name	Manage Books
Actors	Admin
Description	Admin manages the books' details
Pre-conditions	Admin must log in to the system
Main flow	1. Admin navigates to the 'Product' section. 2. System displays the section with the book details. 3. System provide options to add, update and delete a book.
Post-conditions	None

Case name	Add Book
Actors	Admin
Description	Admin adds a new book
Pre-conditions	The use case "Manage Books" must have been executed
Main flow	1. Admin navigates to 'Add Product' sub section. 2. System displays a form to fill. 3. Admin fills the form and sends. 4. System validates the inputs. 6. System adds the book and conforms.
Post-conditions	The new book is added successfully

4.4.3 Financial Module

The function of this module is to manage the Dealers' credits and debits connected to sales and purchases, not to track the financial position of the business. As a result, it does not comply to any standard accounting principles. On-the-spot cash sales in retail.

Case name	View Orders
Actors	Admin
Description	Admin views the orders and transactions
Pre-conditions	Admin must log in to the system
Main flow	1. Admin navigates to the 'Orders' section. 2. System displays the section with the recent orders and transactions details. 3. Admin updates the orders by pending or completed.
Post-conditions	None

4.4.4 Web Module

The Web module is the front-end of the Web-Based System, and it is in charge of making it easier for customers and suppliers to engage with the business. This handles a variety of functions, including making purchases online, viewing order history, and so on.
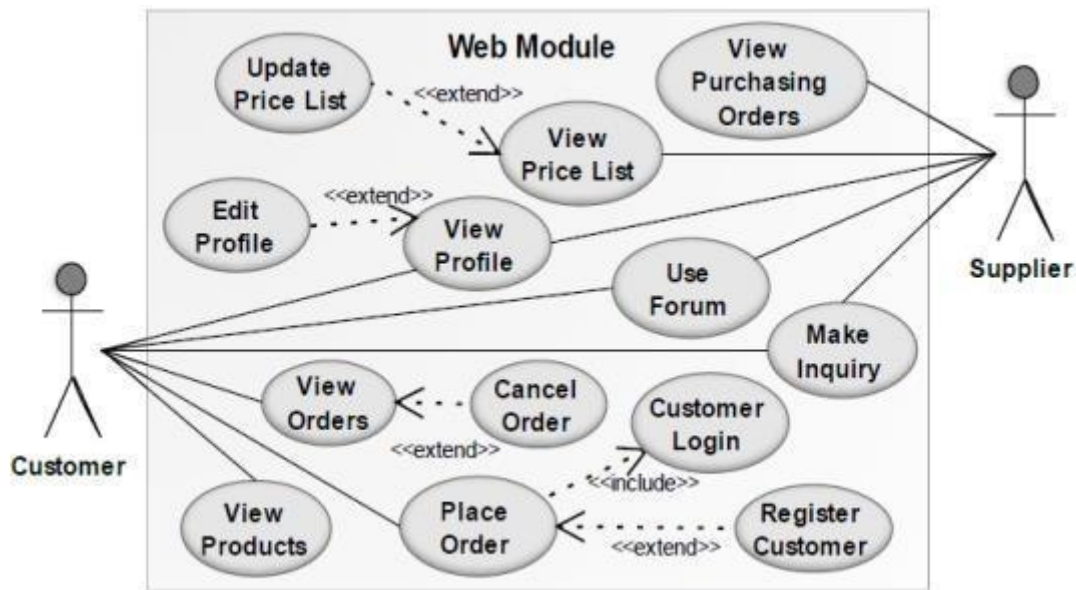
Figure 6: Use case diagram for Web Module

4.4.1 Administration Module

This module is responsible for managing the Web-Based System. This includes managing system users, access levels, system log management, and data backup compilation. If required, the administrator can deactivate or reactivate the users. The data backup is also generated at the administrator's request. Super admin is a virtual administrator who has complete control over all modules. However, it is only recommended that a new administrator be created. Operators and managers are created by the administrator.

Case name	Manage Users
Actors	Admin
Description	Admin updates users
Pre-conditions	Admin must log in to the system
Main flow	1. Admin navigates to the user administration section of the system.

	2. System displays the user's details.
	3. System allows admin to edit, deactivate or reactivate the user based on the user's current status.
	4. System updates the user and confirms.
Post-conditions	None

4.1 Database Design

One of the most important components of any data-driven application, such as a Web-Based System, is the database. As a result, appropriate approaches are used to ensure the database's integrity. MySQL is used as the backend database in this project. MySQL is a database management system that is free and open source. The following are some of MySQL's features:

- MySQL is a database management system that uses a relational model. Rather than one big table, a relational database stores data in multiple tables. These tables can be linked together to make it easier to access and manage data.
- MySQL is a free and open-source database management system. Anyone can use and modify the database software to meet their specific needs. It's quick, dependable, and simple to use to enhance the level of performance.
- MySQL is a database engine that runs in several threads. A multithreaded application accomplishes multiple tasks at once, as if multiple instances of the application were executing at the same time.
- MySQL provides a lot of advantages because it is multithreaded. Each incoming connection is managed by a single thread, with an additional thread that is always running to manage the connections. Multiple clients can perform read operations at the same time, but only one client can access the data being modified while writing. Despite the fact that the threads share the same process space, they operate separately, allowing multiprocessor machines to spread the thread across many CPUs as long as the host

operating system supports multiple CPUs. Multithreading is a critical component for MySQL's performance goals. It is the foundation upon which MySQL is based.

- An ODBC driver is used to link the MySQL database to Java. Open Database Connectivity (ODBC) is a frequently used database access Application Programming Interface (API). The ODBC driver is a library that implements the ODBC API's functions. It handles ODBC function calls, sends SQL requests to the MySQL server, and then returns the results to the application. If necessary, the driver alters an application's request so that it is compatible with MySQL's syntax.

4.2 User Interface Design

One of the most important factors in determining an application's user friendliness is its user interface. Because it is the component with which the user interacts. The following are some of the guidelines that were followed:

- Colors, text styles, component structure, and functionality, such as navigations, should all be consistent across all interfaces.

- All interface effects and dynamic changes should have a clear meaning for the user, such as links with a red pointer cursor for deletion or deletion warning, and so on.

- It should be simple for users to understand. Interfaces should be as basic as possible, with helpful features such as tool tips, popup messages, and notifications recommended.

- Error and confirmation messages, for example, should be consistent, straightforward, and free of technical terms.

- For interfaces such as forms, the data input process should be improved and as many errors as possible detected. It would be helpful to provide default values and watermarks, as well as to use lists and option buttons instead of text boxes for selecting data rather than typing.The sections that follow illustrate some of the

main interfaces and components of the Online Book Store Web System in order to give a sense of the WBBM's overall interfaces.

4.6.1 Register and Login Interface

The system registers and login page, which is part of the developed system, is the primary interface for logging into the system. Both pages are the initial interface a user encounters in any computerised system. As a result, by properly designing and managing errors, the user might develop a positive attitude toward the rest of the system. The main register and login interface for Web-Based Book Store System is shown in Figure 7 and 8.

Chapter 5: IMPLEMENTATION AND EVALUATION

5.1 Introduction

The implementation phase entails the development of an executable program based on the design created during the design phase. Selecting programming languages, additional tools and technologies like as frameworks, selecting hardware platforms, and coding the system are some of the main activities carried out during this phase.

The system is evaluated against multiple factors such as functional and non-functional requirements in order to guarantee that it is working properly and meets all of its specified standards during the evaluation phase of the software development process. This chapter discusses the web-based system testing methodologies, test plan, and test cases, as well as test data and outputs and acceptance testing.

This chapter covers the application's implementation environment, the framework, the development tools used, the application's structure, and an explanation of the key code segments.

5.2 Development Tools and Technologies

- Visual Studio Code for coding.
- XAMPP for local host and server connectivity with MySQL database. Refer to appendix A.
- phpMyAdmin for database manipulation.

The Web-Based Book Store System was built using the following technologies:

- The web pages were developed using Visual Studio Code. The new features that were added in the new version are developed to increase the semantics of web pages. This is the most basic web-related language, and it assists in maintaining a clear and conscious system structure.
- The pages were styled with CSS3. It makes it possible to create a unified design for the entire system in an easy-to-manage manner.
- When developing the system, PHP was employed as the server-side object-oriented programming language.
- The application's database was implemented using MySQL.
- The client-side scripting was implemented with JavaScript. Especially for the form validation process on the client side. It is a scripting language that allows the system components to be continuously changed.
- • JQuery was used to improve the JavaScript development process and to provide various effects on webpages.
- Bootstrap – It's a free framework for developing websites and online apps. It includes design templates that are based on HTML and CSS. With a single code base, Bootstrap easily and efficiently scales the application. Bootstrap speeds up and simplifies frontend web development.

5.3 System Testing

The planning of a testing phase is critical for both the development and completion of a system. The test plan should be able to test the overall system's functionality. By extensively testing a system, it is possible to identify and fix errors that occur as a result of the system. Multiple test scenarios were used to evaluate the implemented system. The test plan has continued to test the system units since the development began. Upon completing the system, it was thorough evaluated to determine whether it could execute as planned. As a result, this testing stage assisted in the early detection of errors.

Following the system unit testing, integration testing was conducted, which allowed for the detection of errors. To evaluate the functionality of the fully developed web-based application, system testing was performed as the final stage.

5.4 Testing Methods

In software engineering, a number of software testing methodologies and styles have been introduced to test various features of various systems. The major testing methodologies utilized widely in the software testing process are Black Box testing and White Box testing. In Black Box testing, the function is evaluated by comparing the output to the input without taking into account the inner structure of the function, whereas in White Box testing, each of the function's logical routes is evaluated while taking into account the inner structure of the function.

Various styles of testing were utilized at various stages of the web-based system's development. During the design phase, unit testing was performed on each individual function of each module to check that they were correct. Following the completion of unit testing, integration testing was performed to check that the functionality of the modules when they interacted with one another was correct. Following the conclusion of the integration testing, system testing was performed on the entire system to guarantee its reliability.

Do refer the results on *Appendix A – Output Screenshot* below.

5.4.1 Purchase Module

The following table list a relevant test case for the Purchase Module.

Test No.	Test Description	Testing Procedures	Expected Result
1.	View purchases / orders	Navigate to Orders section.	Display the sets of recent purchases and placed orders.
2.	Delete purchase order	Click 'Delete Order' button in particular placed order.	Update the placed order as deleted.
3.	Complete purchase order	Click 'Update' button in particular placed order. Fill the relevant details in the dropdown bar as pending or completed and click 'Update' button.	Update the purchase order as confirmed.

5.4.2 Stock Module

The following table list a relevant test case for the Stock Module.

3.	Delete book	Click the 'Delete' button in front of the particular book record and confirm.	Book is deleted from the record.

5.4.3 Financial Module

The following table list a relevant test case for the Financial Module.

Test No.	Test Description	Testing Procedures	Expected Result
1.	View transactions	Navigate to Dashboard section.	Display most recent set of transactions.

5.4.4 Web Module

The following table lists the most relevant test cases for the Web Module.

Test No.	Test Description	Testing Procedures	Expected Result
1.	Main menu navigation	Click each menu item of the main menu.	Load the relevant page.
2.	Place inquiry	Fill in the form in the contact page. Press send button.	Add the inquiry entry & display confirmation.
3.	Customer / admin login	Enter id and password of the customer / admin. Click login.	customer / admin is able to login.
4.	Add books to the shopping cart	Clicks 'Add to Cart' button of the particular product.	Update the shopping cart.

5.1 Test Data and Results

Test data is a collection of information that is used in the testing process. When preparing test data, some of the requirements were carefully followed, such as collecting not only valid but also invalid datasets to cover all parts of the testing process and selecting the smallest data set possible to reduce extra complexity. In unit and integration testing, dummy datasets were utilized, but in Web-Based system testing, datasets retrieved from the current system were used. The test results generated during the execution of the test cases were also documented, which will come in handy for future review and maintenance. Do refer to appendix B – Coding Lists.

5.2 Acceptance Testing

After the system testing was completed, the users were chosen to reflect the WB's whole business operation. After that, all of the input was collected and analyzed, and slight changes were made in response to the users' requests.

User Acceptance Testing was entered into the developed system to begin testing. There were a few small changes that needed to be made to the system. The client tested the entire system by modifying the user's rights. Following the system's testing, it was requested that the system be tested with staff members. Under the instruction of the Administrators, they had tested the system by logging onto their user accounts. The admin pointed up a few minor changes that needed to be made. When they finished their session, they expressed satisfaction with the system. All of the users who recommended minor changes after completing the user acceptance testing gave a positive response. Rather of continuing to use the old way, the client indicated that the newly designed system would allow the business to operate more efficiently and smoothly.

Chapter 6: CONCLUSION AND FUTURE WORK

6.1 Conclusion

Online shopping has gained importance not only from the standpoint of the entrepreneur, but also from the standpoint of the customer, since the Internet has become a vital resource in modern business. Electronic shopping opens up new business prospects for the entrepreneur, and it allows customers to compare prices. According to a report, most online shoppers are impulsive and decide whether or not to stay on a site within the first few seconds. "Website design is similar to the interior design of a store." If the shop appears to be run-down or similar to hundreds of other shops, the customer is likely to move on to the next site." As a result, I created the project to give the user as much ease of navigation, data retrieval, and essential feedback as possible. The user is given with an ecommerce web site that can be utilized to purchase books online in this project.

I utilized PHP to make this a web application. PHP (XAMPP) has a number of advantages, including improved performance, scalability, security, and simplicity. To create a web application with PHP, I'll need a programming language like JAVA, HTML, or anything similar. The language used to develop this application was JAVA. PHP connects with the database using the MySQL server because it offers in-memory caching, which reduces the need to visit the database server frequently, and it is simple to deploy and manage. MySQL was chosen as the back-end database because it is one of the most widely used open-source databases, with rapid data access, ease of installation, and simplicity.

A user-friendly shopping cart functionality must match a solid shopping cart design.

Viewing the contents of the customer's cart and being able to remove or add items to the cart should be easy.

This project's shopping cart application includes a variety of features aimed at making the customer's experience more pleasant. This project will help me understand how to make an interactive web page and the tools that go into making it. The project's architecture, which contains a Data Model and a Process Model, shows how the database is constructed with various tables, as well as how data is accessed and processed from the tables. The project's development has provided me a thorough understanding of how PHP is utilized to develop a website, how it connects to a database to obtain data, and how the data and web pages are modified to provide a shopping cart application to the user.

6.2 Recommendation for Future Work

Software development is a never-ending process that maintains the software's life based on the changing needs of the user throughout time. The project will undoubtedly be designed with easy modification and enhancement in mind, which may be required from time to time. This project, on the other hand, can be modified in a variety of ways. Because of limited a time frame, I am unable to incorporate many things here. But I will attempt to cover all of the existing system that the Online Book Store should consist.

During the development phase, various improvements were found that would be considered in future versions of the Web-Based Online Book Store System:

- Introduce the Barcode System
  Adding an effective Bar Cord Reader helps to improve data capture reliability at the operational level. The accuracy of data entry for sales and purchasing activities can be improved by employing system-generated barcodes for items' SKU

- Provide an SMS service when orders are processed and for special jobs. By implementing this recommendation, users will be able to acquire relevant information from the system without having to enter the system or their email account.
- As the Intranet deals with a considerable amount of sensitive information, improve security by putting in a standardized Firewall and Gateways methods to tighten security even more.
- Obtain an SSL Certificate to ensure that all transactions are conducted over a secure channel using https.
- Make plans to secure an Intellectual Property (IPR) for the company's website.

6.3 Lessons Learnt

As a student participating in a degree program, this was a fantastic opportunity for me to put what I had learned in class into practice. Because the domain area was so extensive, it allowed me to get experience in a variety of business strategies and broadened my understanding of how to map connected business operations into a computerized system. It also gives me an excellent opportunity to put everything I've learned into practice. The creation of the Web-Based Online Book Store System helps in gaining valuable knowledge in system development throughout the SDLC.

Various software engineering approaches, such as requirement collecting techniques, OOD techniques, and designing systems using UML, are practiced during the analysis and design phases, which will be useful in my future job. This phase also assists me in improving my knowledge of PHP, AJAX, JQuery, and other programming languages. The evaluation phase provided a great opportunity for me to practice evaluating systems, particularly on the user side. A better understanding of how to communicate with a real customer was gained. In addition, the working environment included web-related languages.

- Provide an SMS service when orders are processed and for special jobs. By implementing this recommendation, users will be able to acquire relevant information from the system without having to enter the system or their email account.
- As the Intranet deals with a considerable amount of sensitive information, improve security by putting in a standardized Firewall and Gateways methods to tighten security even more.
- Obtain an SSL Certificate to ensure that all transactions are conducted over a secure channel using https.
- Make plans to secure an Intellectual Property (IPR) for the company's website.

6.4 Lessons Learnt

As a student participating in a degree program, this was a fantastic opportunity for me to put what I had learned in class into practice. Because the domain area was so extensive, it allowed me to get experience in a variety of business strategies and broadened my understanding of how to map connected business operations into a computerized system. It also gives me an excellent opportunity to put everything I've learned into practice. The creation of the Web-Based Online Book Store System helps in gaining valuable knowledge in system development throughout the SDLC.

Various software engineering approaches, such as requirement collecting techniques, OOD techniques, and designing systems using UML, are practiced during the analysis and design phases, which will be useful in my future job. This phase also assists me in improving my knowledge of PHP, AJAX, JQuery, and other programming languages. The evaluation phase provided a great opportunity for me to practice evaluating systems, particularly on the user side. A better understanding of how to communicate with a real customer was gained. In addition, the working environment included web-related languages.

REFERENCES

☐ L. T. T. Tran (2021, January). Managing the effectiveness of e-commerce platforms in a pandemic. https://www.sciencedirect.com/science/article/pii/S0969698920312959

☐ Essay Sauce (2019, August). Analyze the website requirements both functional and non- functional of the XYZ bookstore. https://www.essaysauce.com/information-technology-essays/analyze-the-website-requirements-both-functional-and-non-functional-of-the-xyz-bookstore/

☐ Shirley R. (2009, April). The System Development Life Cycle (SDLC). https://csrc.nist.gov/csrc/media/publications/shared/documents/itl-bulletin/itlbul2009- 04.pdf

☐ Gaurav K. (2012, August). Impact of Agile Methodology on Software Development Process. https://www.researchgate.net/profile/Gaurav-Kumar-175/publication/255707851_Impact_of_Agile_Methodology_on_Software_Developmen t_Process/links/00b49520489442e12d000000/Impact-of-Agile-Methodology-on- Software-Development-Process.pdf

☐ Verma A. (2017, November). A Comparative Study of Black Box Testing and White Box Testing. https://www.researchgate.net/profile/Sarika-Chaudhary/publication/325816726_A_Comparative_Study_of_Black_Box_Testing_and_

- Leonidio, U. da C., Montezano, R. M. da S., Carvalho, F. A. de (2017, November).

  Evaluation of Perceived Quality of The Website of An Online Bookstore: An Empirical Application of The Barnes and Vidgen Model.
  https://www.scielo.br/j/jistm/a/cHf5LJhTf7sQnByJn5zgSPp/?format=pdf&lang=en

- Huang, L. C. (2017, November). Customer Relationship Management (CRM) in

  Business- to-Business (B2B) e-commerce.
  https://www.emerald.com/insight/content/doi/10.1108/IntR-05-2016-0142/full/html

- Hsu, C.L., Wu, C.C. and Chen, M.C. (2013, April). How social shopping retain

  customers? Capturing the essence of website quality and relationship quality
  https://www.researchgate.net/publication/299442291_How_social_shopping_retain
  _c ustomers_Capturing_the_essence_of_website_quality_and_relationship_quality

APPENDIX A – OUTPUT SCREENSHOT
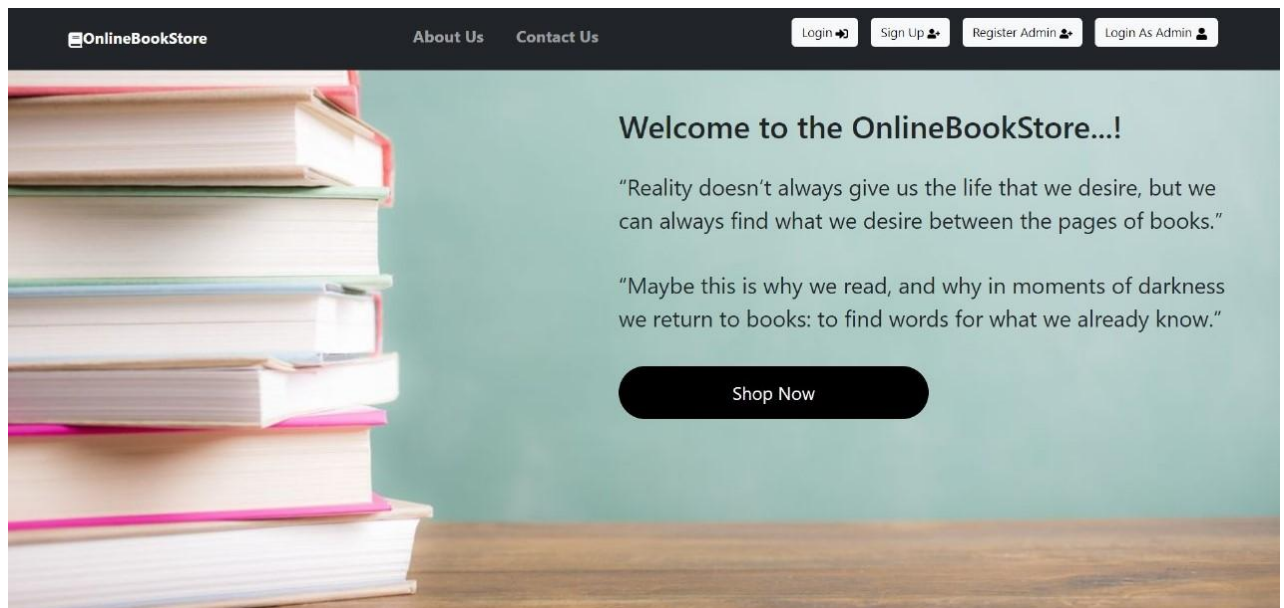
Figure : Welcome Page
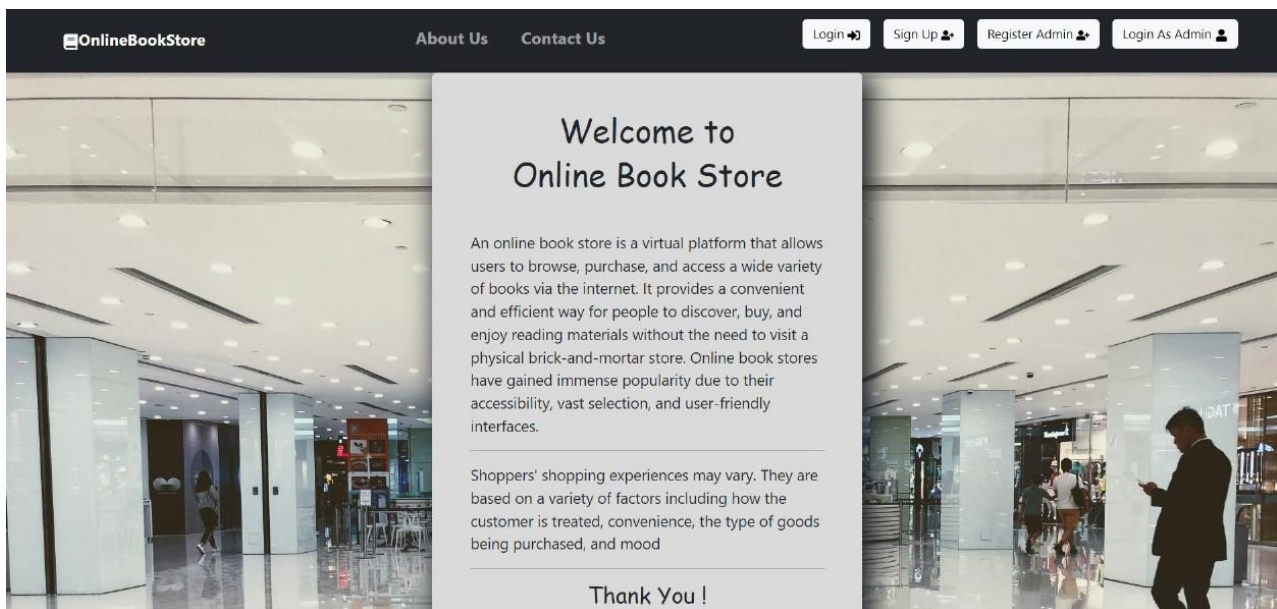
Figure : Welcome Page After Login

Figure : Admin Registration Page

Figure : Admin Login Page
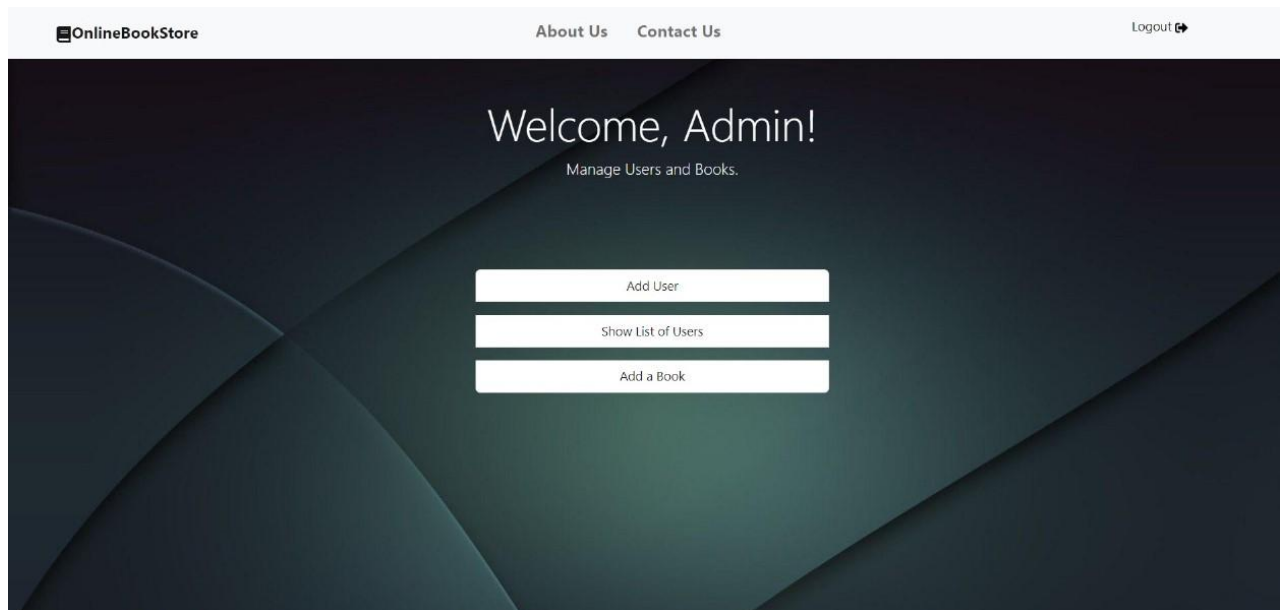
Figure : Amin Welcome Page

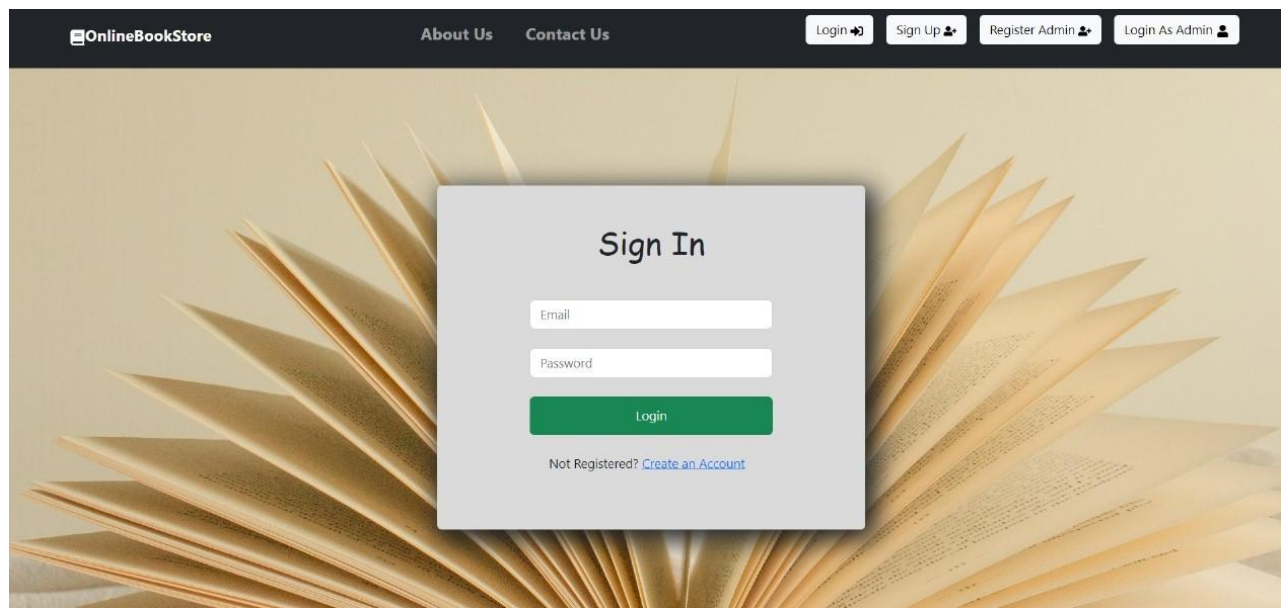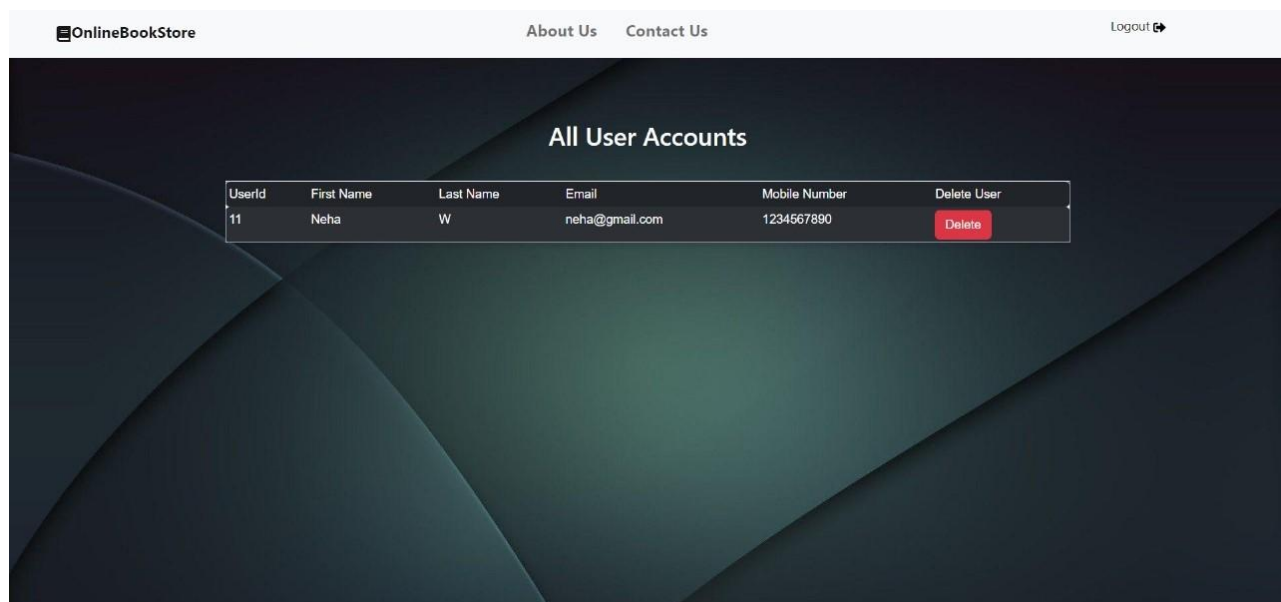*Figure : User Registration Page*

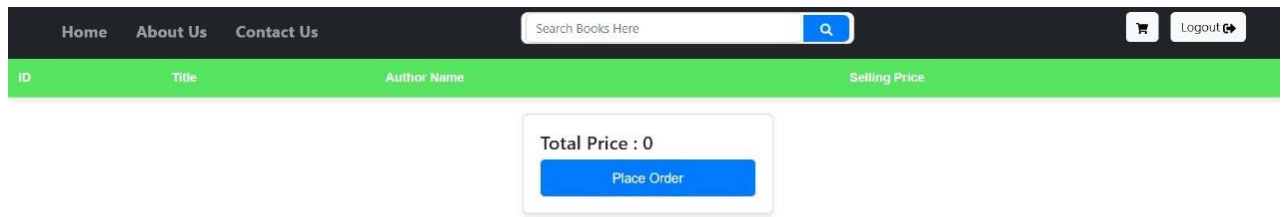*Figure : User Sign in Page*

Figure : All User Details
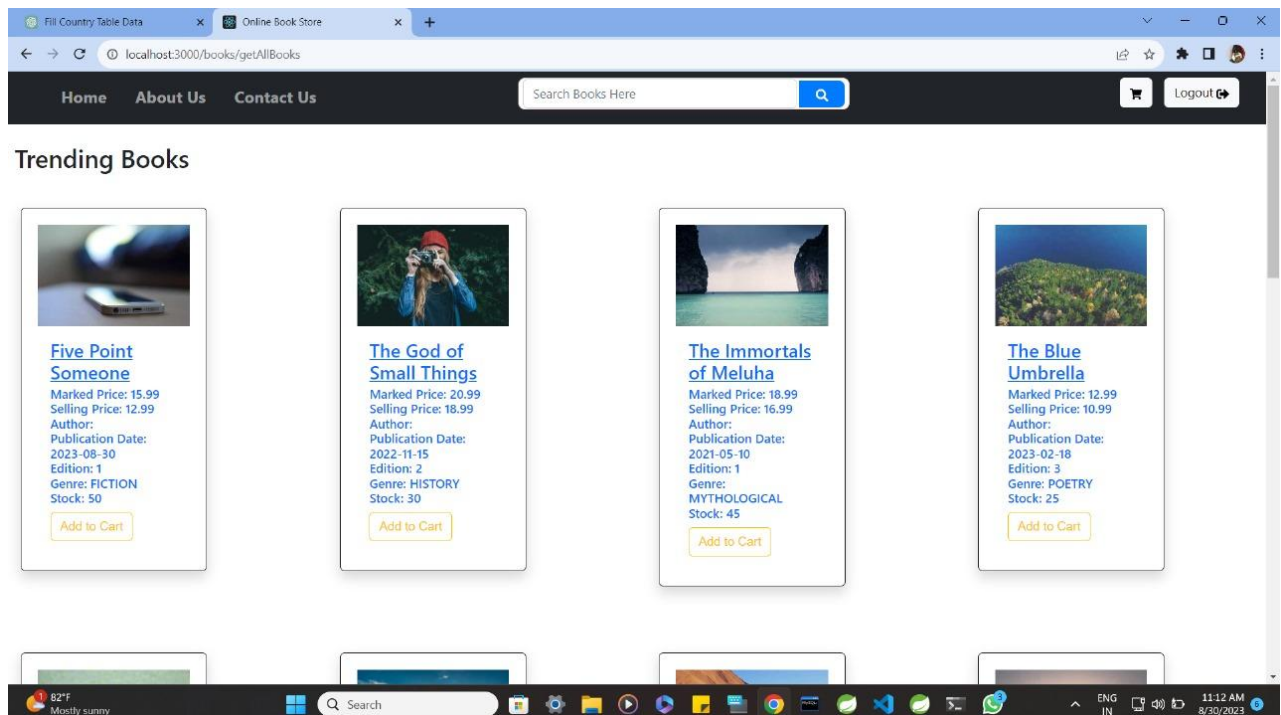
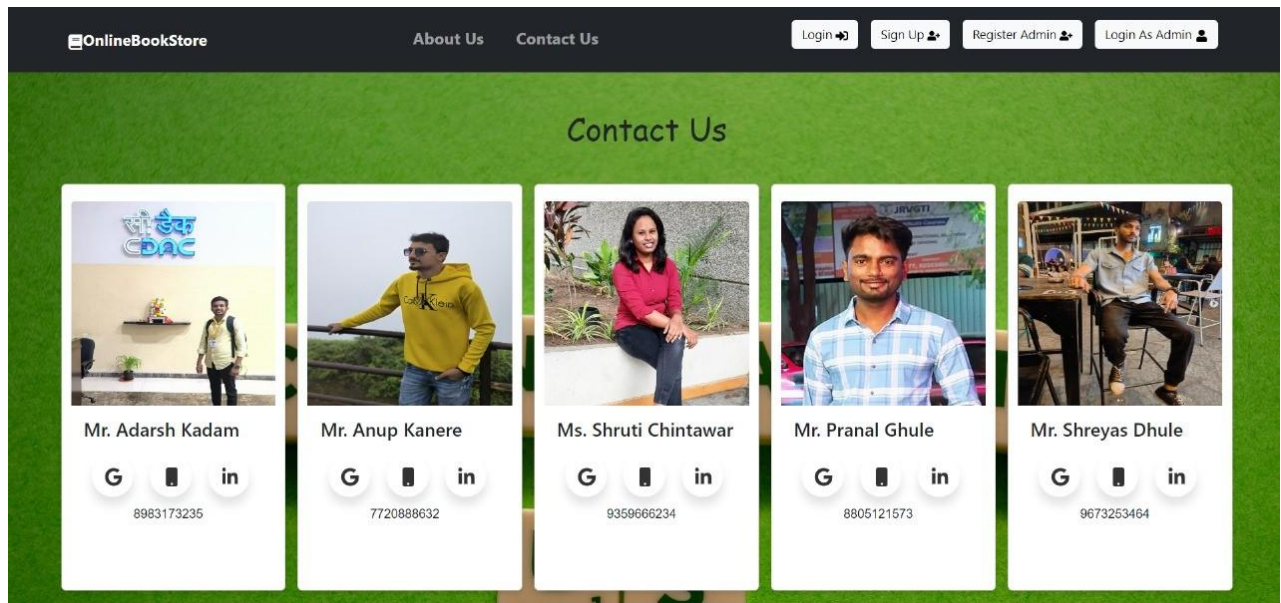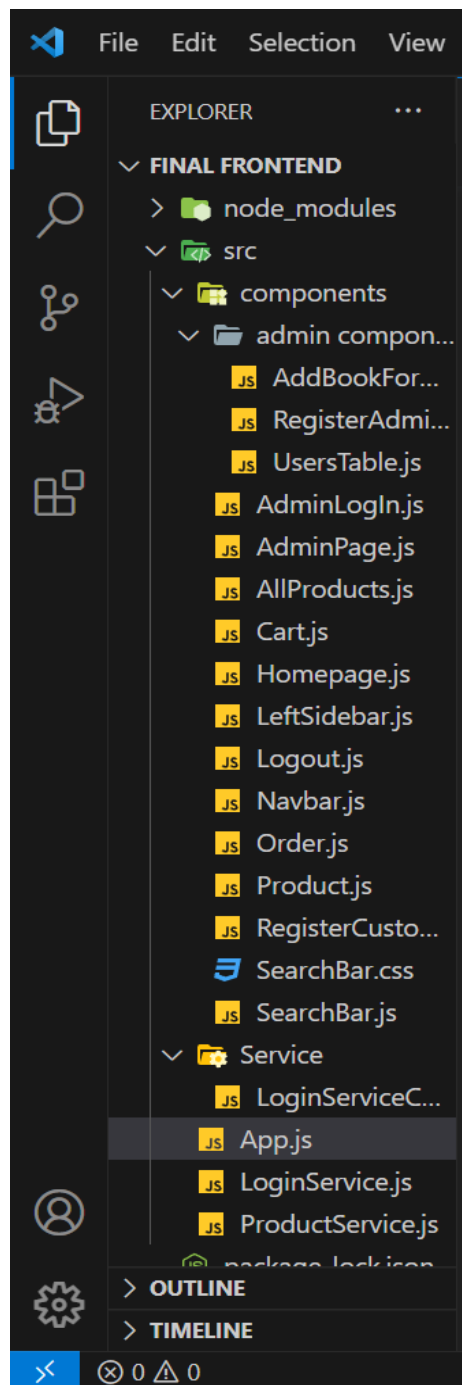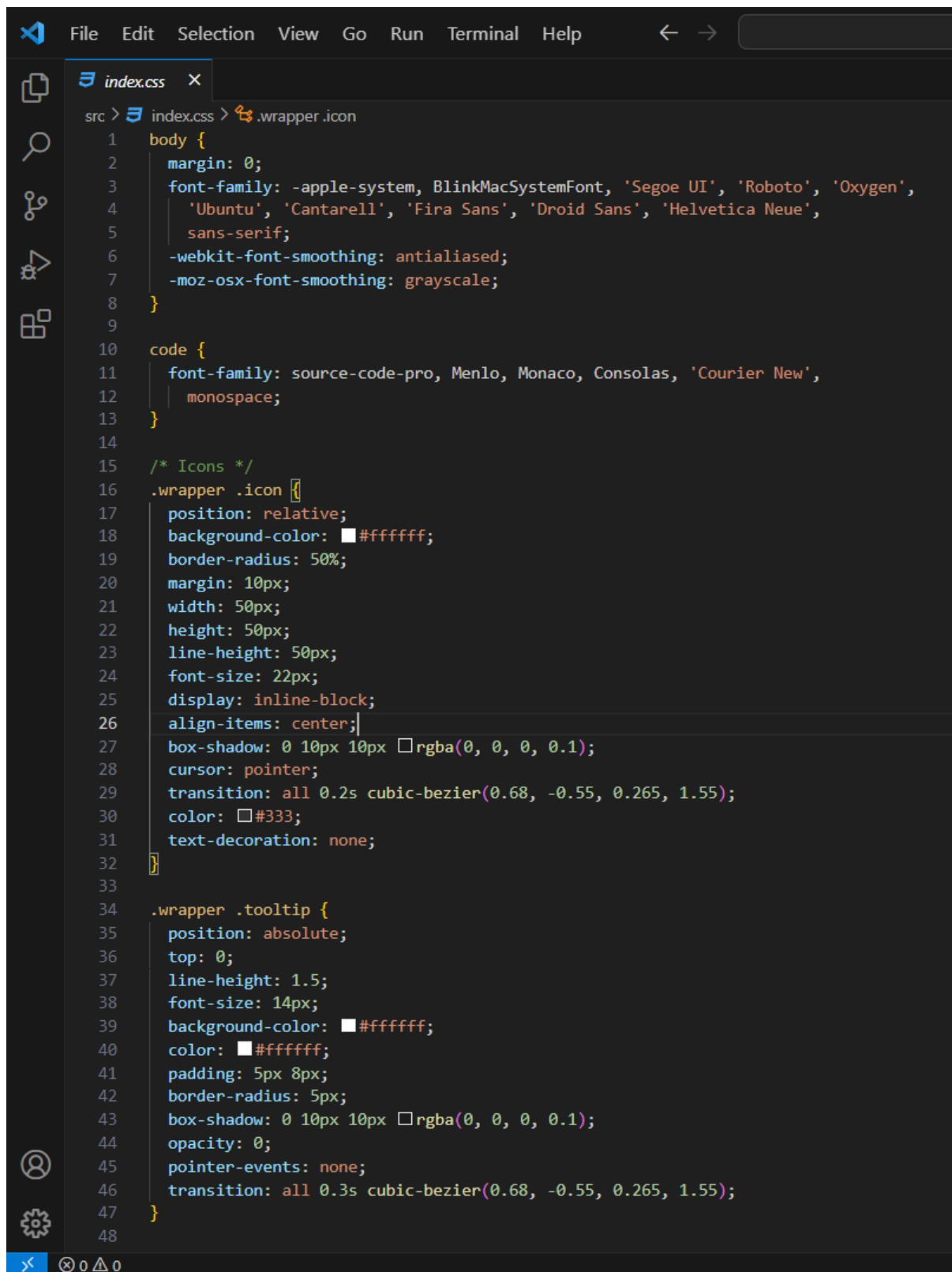*Figure : Adding and Placing Order*

Figure : Trending Book Page

Figure : Contact Us Page

APPENDIX B - CODING LISTS

This part contains several Web-Based Book Store System codes that has been explained in the design and implementation chapter.

General components of the program

```
index.css   ×

src > 3 index.css > & .wrapper .icon
  1    body {
  2      margin: 0;
  3      font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', 'Roboto', 'Oxygen',
  4        'Ubuntu', 'Cantarell', 'Fira Sans', 'Droid Sans', 'Helvetica Neue',
  5        sans-serif;
  6      -webkit-font-smoothing: antialiased;
  7      -moz-osx-font-smoothing: grayscale;
  8    }
  9
 10    code {
 11      font-family: source-code-pro, Menlo, Monaco, Consolas, 'Courier New',
 12        monospace;
 13    }
 14
 15    /* Icons */
 16    .wrapper .icon {
 17      position: relative;
 18      background-color: #ffffff;
 19      border-radius: 50%;
 20      margin: 10px;
 21      width: 50px;
 22      height: 50px;
 23      line-height: 50px;
 24      font-size: 22px;
 25      display: inline-block;
 26      align-items: center;
 27      box-shadow: 0 10px 10px rgba(0, 0, 0, 0.1);
 28      cursor: pointer;
 29      transition: all 0.2s cubic-bezier(0.68, -0.55, 0.265, 1.55);
 30      color: #333;
 31      text-decoration: none;
 32    }
 33
 34    .wrapper .tooltip {
 35      position: absolute;
 36      top: 0;
 37      line-height: 1.5;
 38      font-size: 14px;
 39      background-color: #ffffff;
 40      color: #ffffff;
 41      padding: 5px 8px;
 42      border-radius: 5px;
 43      box-shadow: 0 10px 10px rgba(0, 0, 0, 0.1);
 44      opacity: 0;
 45      pointer-events: none;
 46      transition: all 0.3s cubic-bezier(0.68, -0.55, 0.265, 1.55);
 47    }
 48
```

⊗ 0 ⚠ 0

index.css

App.js

src > App.js > ...

```js
import './App.css';
import { ToastContainer } from "react-toastify";
import AdminPage from './components/AdminPage';
import { BrowserRouter, Routes, Route } from 'react-router-dom';
import UsersTable from './components/admin components/UsersTable';
import RegisterAdmin from './components/admin components/RegisterAdmin';
import Homepage from './components/Homepage';
import Contact from './components/Contact';
import About from './components/About';
import LoginForm from './components/LoginForm';
import RegisterCustomer from './components/RegisterCustomer';
import Cart from './components/Cart';
import Order from './components/Order';
import AdminLogIn from './components/AdminLogIn';
import AddBookForm from './components/admin components/AddBookForm';
import AllBooks from './components/AllProducts';
import Logout from './components/Logout';
import NotFound from './components/NotFound';
import UpdateUser from './components/UpdateUser';


function App() {

  return (

    <BrowserRouter>
      <Routes>
        <Route exact path="/" element={<Homepage />} />
        <Route exact path="/users/register-admin" element={<RegisterAdmin />} />
        <Route exact path="/users/admin-login" element={<AdminLogIn />} />
        <Route exact path="/admin" element={<AdminPage />} />
        <Route exact path="/users/register-customer" element={<RegisterCustomer />} />
        <Route exact path="/users" element={<UsersTable />} />
        <Route exact path="/books/getAllBooks" element={<AllBooks />} />
        <Route exact path="/books/add-book" element={<AddBookForm />} />
        <Route exact path="/UpdateUser" element={<UpdateUser />} />
        <Route exact path="/users/signin" element={<LoginForm />} />
        <Route exact path="/contact" element={<Contact />} />
        <Route exact path="/about" element={<About />} />
        <Route exact path="/carts/getBooksInCart" element={<Cart />} />
        <Route exact path="/myorder" element={<Order />} />
        <Route exact path="/logout" element={<Logout />} />
        <Route exact path="*" element={<NotFound />} />

      </Routes>
      <ToastContainer
        position="bottom-center"
        autoClose={1000}
      ></ToastContainer>
    </BrowserRouter>
  );
}


export default App;
```

App.js

LoginForm.js ×

src > components > LoginForm.js > ...

```javascript
1   import Navbar from "./Navbar";
2   import "../App.css";
3   import { useState } from "react";
4   import { useNavigate } from "react-router-dom";
5   import LoginServiceCustomer from "../Service/LoginServiceCustomer";
6
7   function LoginForm() {
8
9       const [userId, setuserId] = useState(0);
10      const navigate = useNavigate();
11      const [email, setEmail] = useState('');
12      const [password, setPassword] = useState('');
13      const [status, setStatus] = useState('');
14      const handleSubmit = (event) => {
15          event.preventDefault();
16          if (email === '' || password === '') {
17              setStatus('Enter email id and password')
18          } else {
19              LoginServiceCustomer.validateuser(email, password).then((result) => {
20                  console.log(email);
21                  var msg = JSON.stringify(result.data.message);
22                  console.log(msg);
23                  console.log(result);
24                  setuserId(userId);
25                  sessionStorage.setItem('userId', result.data.userDTO.userId);
26                  sessionStorage.setItem('email', email);
27                  sessionStorage.setItem("user", (result.data.userDTO));
28                  navigate("/books/getAllBooks");
29
30              }).catch((err) => { setStatus('Invalid email or password..!') });
31          }
32      }
33
34      return (
35          <div>
36              <Navbar></Navbar>
37
38              <div className="container-fluid login-form " >
39                  <form onSubmit={handleSubmit}>
40                      <div className="row align-items-center justify-content-evenly " style={{ height: "93vh" }}>
41                          <div className="col-4 align-self-center p-5 rounded">
42                              <h1 className="log-in-in mb-5">Sign In</h1>
43
44
45                              <div className="row m-4 align-items-center justify-content-evenly">
46                                  <div className="col-10">
47                                      <input className="form-control form-control " type="email" placeholder="Email" value={email} onChange={(event) => setEmail(event.target.value)} required></input>
48                                  </div>
49                              </div>
50
51                              <div className="row m-4 align-items-center justify-content-evenly">
52                                  <div className="col-10">
53                                      <input className="form-control form-control " type="password" placeholder="Password" value={password} onChange={(event) => setPassword(event.target.value)} required></input>
54                                  </div>
55                              </div>
56
57                              <div className="row m-4 align-items-center justify-content-evenly">
58                                  <div className="col-10">
59                                      <button className="btn btn-success rounded" type="submit">Login</button>
60                                      {status ? <div className='text-success'>{status}</div> : null}
61                                  </div>
62                              </div>
63
64                              <div className="row m-4 align-items-center justify-content-evenly">
65                                  <div className="col-10">
66                                      <span className="lastspan" >Not Registered? <a href="/users/register-customer" className="link-primary">Create an Account</a></span>
67                                  </div>
68                              </div>
69
70                          </div>
71
72                      </div>
73
74                  </form>
75              </div>
76
77          )
78      )
79  }
80
81  export default LoginForm;
```

LoginForm.js

```js
import React, { useState, useEffect } from "react";
import { Row, Col } from "reactstrap";
import Product from "./Product";
import { FaShoppingCart, FaSignOutAlt } from 'react-icons/fa';
import Navbar from "./Navbar";
import SearchBar from "./SearchBar"
import LoginServiceCustomer from "../Service/LoginServiceCustomer";

function AllBooks() {
  const [products, setProducts] = useState([]);

  useEffect(() => {
    fetchBooks();
  }, []);

  const fetchBooks = async () => {
    try {
      const response = await LoginServiceCustomer.getAllBooks();
      setProducts(response.data);
      console.log(response.data);
    } catch (error) {
      console.error("Error fetching books:", error);
    }
  };

  let user = sessionStorage.getItem("user");

  return (
    <div>
      <nav className="navbar navbar-expand-lg navbar-dark bg-dark" style={{ fontWeight: "bold" }}>
        <div className="container-fluid" style={{ marginLeft: "3rem" }}>
          <div className="collapse navbar-collapse" id="navbarSupportedContent">
            <ul className="navbar-nav me-auto mb-2 mb-lg-1">
              <li className="nav-item">
                <a className="nav-link" aria-current="page" href="/" style={{ fontSize: "1.3rem", marginRight: "1.3rem" }}>Home</a>
              </li>

              <li className="nav-item">
                <a className="nav-link" href="/about" style={{ fontSize: "1.3rem", marginRight: "1.3rem" }}>About Us</a>
              </li>
              <li className="nav-item">
                <a className="nav-link" href="/contact" style={{ fontSize: "1.3rem" }}>Contact Us</a>
              </li>
              <div style={{ marginLeft: "15rem" }}>

                <SearchBar />
              </div>

              <div className='buttons' style={{ marginLeft: "20rem" }}>
                <a href='/carts/getBooksInCart' className='btn btn-light ms-3'>
                  <FaShoppingCart />
                </a>
                <a href='/logout' className='btn btn-light ms-3'>
                  Logout <FaSignOutAlt />
                </a>
              </div>
            </ul>
          </div>
        </div>
      </nav>

      <h2 className="ms-2 my-4">Trending Books</h2>

      <Row>
        {products.length > 0 ? (
          products.map((item) => (
            <Col sm="3" key={item.title}>
              <Product product={item} />
            </Col>

          ))
        ) : (
          <p>No Products under this category</p>
        )}
      </Row>
      <p></p>
    </div>
  );
}

export default AllBooks;
```

AllProducts.js

DB cities

DBCountries

DB Tables

DB Languages

```
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use online_book_store;
Database changed
mysql> select * from books;
+---------+----------------------------+------------+----------------------------------------------------------------+---------+--------------+--------------+--------------+------------------+--------------+-------+--------------------------+
| book_id | author                     | book_image | description                                                    | edition | genre        | is_available | marked_price | publication_date | selling_price | stock | title                    |
+---------+----------------------------+------------+----------------------------------------------------------------+---------+--------------+--------------+--------------+------------------+--------------+-------+--------------------------+
|       1 | Chetan Bhagat              | image1.jpg | A story about young love and ambition.                         |       1 | FICTION      | 0x01         |        15.99 | 2023-08-30       |        12.99 |    50 | Five Point Someone       |
|       2 | Arundhati Roy              | image2.jpg | A poignant tale set in post-colonial India.                   |       2 | HISTORY      | 0x01         |        20.99 | 2022-11-15       |        18.99 |    30 | The God of Small Things  |
|       3 | Amish Tripathi             | image3.jpg | A mythological reimagining of Shiva and his adventures.        |       1 | MYTHOLOGICAL | 0x01         |        18.99 | 2021-05-10       |        16.99 |    45 | The Immortals of Meluha  |
|       4 | Ruskin Bond                | image4.jpg | Stories capturing the beauty of the Himalayan region.         |       3 | POETRY       | 0x01         |        12.99 | 2023-02-18       |        10.99 |    25 | The Blue Umbrella        |
|       5 | Jhumpa Lahiri              | image5.jpg | Exploring the lives of Indian immigrants in the United States. |       1 | DRAMA        | 0x01         |        16.99 | 2022-06-05       |        14.99 |    40 | Interpreter of Maladies  |
|       6 | Vikram Seth                | image6.jpg | A sweeping family saga spanning different generations and continents. | 2 | FICTION  | 0x01         |        22.99 | 2021-09-20       |        20.99 |    20 | A Suitable Boy           |
|       7 | R.K. Narayan               | image7.jpg | Whimsical stories set in the fictional town of Malgudi.       |       1 | FICTION      | 0x01         |        14.99 | 2022-03-12       |        11.99 |    60 | Swami and Friends        |
|       8 | Chitra Banerjee Divakaruni | image8.jpg | A tale of friendship and sacrifice among four women in India. |       2 | FICTION      | 0x01         |        18.99 | 2023-07-28       |        16.99 |    35 | The Palace of Illusions  |
|       9 | Rabindranath Tagore        | image9.jpg | Collection of poems and stories by the Nobel laureate.        |       1 | POETRY       | 0x01         |        10.99 | 2021-11-02       |         8.99 |    50 | Gitanjali                |
|      10 | Vikram Chandra             | image10.jpg| A gripping crime novel set in the city of Mumbai.             |       1 | THRILLER     | 0x01         |        20.99 | 2023-04-17       |        18.99 |    30 | Sacred Games             |
|      11 | Khushwant Singh            | image11.jpg| A collection of satirical and humorous short stories.         |       2 | COMIC        | 0x01         |        15.99 | 2022-09-08       |        12.99 |    45 | The Mark of Vishnu       |
|      12 | Anuja Chauhan              | image12.jpg| A romantic novel revolving around cricket and love.           |       1 | ROMANCE      | 0x01         |        16.99 | 2021-12-05       |        14.99 |    25 | The Zoya Factor          |
|      13 | Raja Rao                   | image13.jpg| An exploration of the spiritual journey of an Indian man.     |       1 | FICTION      | 0x01         |        18.99 | 2022-01-23       |        15.99 |    20 | Kanthapura               |
|      14 | Sudha Murty                | image14.jpg| Heartwarming stories inspired by real-life experiences.       |       1 | FICTION      | 0x01         |        14.99 | 2023-06-11       |        12.99 |    40 | Wise and Otherwise       |
|      15 | Shashi Tharoor             | image15.jpg| A historical novel set against the backdrop of the Partition. |       2 | HISTORY      | 0x01         |        22.99 | 2021-08-30       |        19.99 |    15 | The Great Indian Novel   |
+---------+----------------------------+------------+----------------------------------------------------------------+---------+--------------+--------------+--------------+------------------+--------------+-------+--------------------------+
15 rows in set (0.00 sec)

mysql>
```

DataBase FetchData